From Treebank Metadata to Sentence-Level Genre in Universal Dependencies: A Reproducible, Versioned Resource

Egon W. Stemle
Institute for Applied Linguistics, Eurac Research
Faculty of Informatics, Masaryk University
egon.stemle@eurac.edu

Abstract

We release a sentence-level genre layer for Universal Dependencies as a separate, joinable dataset, computed across UD revisions and linked back to the underlying treebanks via a release-aware composite key comprising treebank, split, `sent_id`, and UD release metadata. The annotations are derived rather than authoritative and are accompanied by provenance and uncertainty indicators, enabling downstream users to choose appropriate precision-coverage trade-offs and to re-run the pipeline as UD evolves. To support both parity tracking and deployment-oriented interpretation, we report results under two complementary regimes: a fixed-partition setting aligned with earlier protocols, and a language-grouped 10-fold generalisation setting that highlights cross-language heterogeneity and anchor sparsity as operational constraints. The resulting resource is intended to make genre a practical control variable for UD-based experimentation, including genre-stratified evaluation and training data selection for POS tagging and parsing, where performance varies substantially across text types. Finally, we note that reduced genre spaces aligned with recurring robustness profiles (e.g. transcribed speech versus interactional web/social text versus edited prose/news) appear pragmatically useful, but should be treated as a community coordination task implemented through explicit, versioned mapping tables.

Keywords: sentence-level genre annotation, cross-lingual genre inference, linguistic data infrastructure

1. Introduction

Universal Dependencies (UD) is a central multilingual resource for morphosyntactic annotation, but its genre metadata remains coarse: genres are declared at treebank level, often multi-valued, and usually not linked to individual sentences (Nivre et al., 2020; Müller-Eberstein et al., 2021b; Danilova and Stymne, 2023). This is limiting whenever treebanks mix edited prose, interactional Web text, speech-like material, or learner writing, because treebank identity then becomes a poor proxy for the text conditions that matter in downstream experiments.

Sentence-level genre is useful for at least two practical reasons. First, it supports genre-stratified evaluation and error analysis in POS tagging, parsing, and related tasks, where performance can differ sharply between edited and non-edited text (Giesbrecht and Evert, 2009; Owoputi et al., 2013a; Behzad and Zeldes, 2020). Second, it supports training-data selection and cross-genre transfer, including settings where in-genre or genre-aware proxy data is more informative than simply choosing data by language or size (Rehbein and Bildhauer, 2017; Müller-Eberstein et al., 2021a).

Two existing lines of work frame the present paper. Müller-Eberstein et al. show that sentence-level genre can be weakly recovered from UD via multilingual embeddings, treebank-local clustering, and metadata-driven labelling (Müller-Eberstein et al., 2021b). UD-MULTIGENRE shows that careful manual reconstruction can yield high-quality instance-level labels, but only for a limited subset of UD (Danilova and Stymne, 2023). The resulting gap is infrastructural: weak supervision scales, manual curation is precise, but neither by itself yields a persistent, versioned genre layer for UD as a whole.

We address that gap by releasing sentence-level genre as a separate sidecar dataset rather than modifying UD treebanks directly. The layer is linked back to UD via a composite sentence reference over treebank, split, `sent_id`, and release metadata; it is versioned across UD releases; and it is generated by a reproducible pipeline that combines metadata extraction with bootstrapped clustering. Alongside the labels, we publish provenance and uncertainty information so that consumers can trade off coverage and precision explicitly.

Our contribution is therefore twofold: (i) a full-coverage, reproducible sentence-level genre layer for UD treebanks `hf://datasets/commul-/ud_genre` (Stemle, 2026a), and (ii) the supporting infrastructure `hf://datasets/commul-/universal_dependencies` (Stemle, 2026b) and `ud-hf-parquet-tools` (Stemle, 2026e) that makes this layer regenerable and inspectable across releases. The paper also distinguishes paper-style comparability from end-user sentence-level evaluation, since these optimise different targets.

Proceedings of the Ninth Workshop on Universal Dependencies (UDW 2026), pages 268–276
May 16, 2026.

2. Background and Related Work

2.1. Genre Information in Universal Dependencies

UD's genre inventory includes familiar labels such as *news*, *wiki*, *blog*, *email*, *spoken*, *fiction*, and *legal*, but the values mix medium, domain, communicative situation, and editorial status, as the Universal Dependencies contributors (2026) acknowledge. The labels are assigned at treebank level, often multi-valued, and there is no standard machine-readable mechanism for attaching them to individual sentences. For multi-genre treebanks this leaves the internal genre distribution largely implicit.

2.2. Weakly Supervised and Curated Sentence-Level Genre

Müller-Eberstein et al. provide the main weakly supervised reference point: sentence embeddings, treebank-local clustering, and metadata-driven labelling can recover sentence-level genre signals at scale, and local clustering is crucial because multilingual embeddings otherwise reflect language more strongly than genre (Müller-Eberstein et al., 2021b). This line of work scales to the full UD collection, but has mainly been distributed as experimental code rather than as a persistent data artefact.

UD-MULTIGENRE (Danilova and Stymne, 2023) takes the complementary route: manual analysis of documentation and metadata yields high-confidence instance-level labels for a substantial but still partial UD subset. It is therefore highly valuable as a curated reference, but not by itself a full-coverage solution.

2.3. Positioning of the Present Work

Our work sits between these two strands. We do not introduce a new genre taxonomy, and we do not aim to replace manually curated resources. Instead, we operationalise sentence-level genre as a versioned, reproducible sidecar layer for UD, built from release-preserving data access, auditable metadata extraction, and weakly supervised extrapolation. In that sense, the contribution is primarily infrastructural: making existing genre-recovery ideas reusable as data, not only as one-off experiments.

3. Methodology

Our methodological point of departure is the observation that genre information in Universal Dependencies (UD) is, in practice, both valuable and structurally under-specified at the instance level: UD treebanks declare which genres are present, yet the prevailing representation remains an inventory at the treebank level, without a standardised machine-readable mechanism to associate individual sentences with genre labels (Universal Dependencies contributors, 2026). In this context, sentence-level genre can be understood as a missing intermediate layer between (i) community-facing corpus descriptions and (ii) infrastructure-facing data access patterns that increasingly assume selective, instance-level filtering.

Against this background, we operationalise a framework that derives sentence-level genre annotations for all UD treebanks across multiple UD releases, while keeping the primary UD data unchanged. Concretely, we treat sentence-level genre labels as a *sidecar layer* that is linked back to UD sentences via a composite key over treebank, split, `sent_id`, and release metadata, and is published as a separate dataset. This separation is not only a pragmatic distribution choice, but it also reflects a layered responsibility model: UD remains the authoritative source for syntactic annotation, whereas derived, task-facing enrichments can evolve in parallel, with explicit provenance and uncertainty signalling.

3.1. Data Basis and Versioned Access

3.1.1. UD Snapshots as Infrastructure Objects

The empirical basis of the framework is a revision-preserving packaging of UD treebanks in Parquet format, exposed as a dataset repository on the Hugging Face Hub. The repository distinguishes explicitly between *framework versioning* (tooling and schema evolution) and *UD data versioning* (the linguistic release snapshots), and it maintains a separate branch for each UD data release. In addition, the dataset repository carries a `tools/` subdirectory with regeneration and inspection scripts so that the published Parquet snapshots are not only downloadable but reproducible from the underlying UD sources. This design makes it possible to compute sentence-level genre annotations consistently for a defined range of UD releases (in our current setup: 2.7 through 2.17), and to keep release-specific inventories intact rather than implicitly normalising across versions. (Stemle, 2026b)

The practical motivation for this choice is that genre inventories, treebank composition, and metadata conventions are not stable across UD releases. A methodology that collapses these differences too early risks conflating (i) genuine linguistic or corpus shifts with (ii) release-driven annotation or documentation changes. Preserving release-specific snapshots, while enabling explicit cross-version mappings, supports both reproducibility and diag-

nostic clarity.

Fidelity-preserving conversion to Parquet. A second infrastructural dependency is a conversion and validation toolchain that generates Parquet representations from CoNLL-U and validates them against the original inputs with explicit fidelity checking (Stemle, 2026e). We deliberately keep this toolchain in the separate `ud-hf-parquet-tools` repository, because the Hugging Face dataset repository is the right place to version released data artefacts, whereas iterative parser, schema, and validation development benefits from a conventional software repository. The tooling is designed to preserve information that is often treated as peripheral but becomes central for metadata-based enrichment, including ordered comment blocks (with duplicates), edge cases in CoNLL-U parsing, and special node types such as multi-word tokens and empty nodes. In particular, the schema exposes `sent_id`, split identity, and preserves sentence-level comments as an ordered list, all of which are prerequisites for robust genre extraction under heterogeneous metadata practices.

From a methodological perspective, this step is not merely an implementation detail. Comment lines and documentation-derived conventions largely mediate genre extraction in UD. Consequently, a conversion step that drops, normalises, or de-duplicates comment metadata would directly bias both coverage estimates and evaluation outcomes.

3.1.2. Genre label space and cross-version comparability

Canonical target labels: UD global genres. For the canonical label space, we primarily adopt UD global genres as defined by UD's genre documentation and decision list. This choice aligns the sidecar annotations with an existing community-facing taxonomy, and it retains a clear conceptual anchor: the goal is not to introduce a competing genre scheme, but to make an existing UD metadata dimension operational at the sentence level.

At the same time, UD's documentation makes visible that the taxonomy has evolved, including explicit changes to formerly permitted genre values and subsequent consolidations. As a consequence, cross-version comparability cannot be treated as a trivial join on string labels; instead, it requires an explicit mapping layer that acknowledges both label drift and treebank-specific conventions.

Mapping tables as a mediation layer. We therefore distinguish between:

1. **Release-specific inventories**, i.e. the set of genre descriptors and metadata signals observed in a given UD release snapshot (kept intact per branch), and
2. **Canonical genres**, i.e. a normalised target set used for cross-treebank and cross-version aggregation.

In the present framework, the mapping layer is seeded from the `tb-genres.json` resource distributed with Müller-Eberstein (2021), which provides mappings from treebank-specific genre labels to global UD genre labels for a subset of treebanks.

However, a more comprehensive, version-aware mapping table is likely needed for long-term maintenance. In practice, this suggests a community-maintainable artefact that (i) records release-local genre strings and extraction cues, (ii) maps them to canonical UD genres for that release, and (iii) optionally provides a second mapping to a stable comparison layer spanning multiple UD versions. The intended contribution of our framework is to make such mediation explicit and testable, rather than leaving it implicit in ad hoc downstream preprocessing.

3.1.3. Metadata-based sentence-level genre extraction

Heterogeneous metadata as an empirical constraint. UD treebanks are heterogeneous not only linguistically but also in their metadata practices. The UD documentation notes that there is no standardised machine-readable way to identify which sentences belong to which genre, and sketches sentence-id ranges as a possible, but not standardised, mechanism. In operational terms, this means that sentence-level genre signals are typically encoded in treebank-specific comment conventions (e.g. document identifiers, source references, partition markers, or informal metadata keys).

Our extraction component treats this heterogeneity as the default case. Instead of assuming one universal comment key, we implement treebank-aware extraction patterns, organised as a configuration layer that can express both:

- **direct genre indicators** (explicit per-sentence labels, where available), and
- **indirect indicators** (document-level or source-level metadata that can be propagated to sentences via structural anchors, such as document boundaries).

Pattern testing and coverage diagnostics. To make extraction patterns maintainable and reviewable, we provide explicit diagnostics functionality in

the genre bootstrapping toolkit. In practice, these commands are part of the release workflow rather than ad hoc debugging utilities, because each promoted release is preceded by coverage checks and treebank-level spot tests:

- `test-genres`, which supports iterative testing and debugging of extraction patterns, and surfaces coverage statistics to quantify how much of a treebank is addressable by the current extraction rules (Stemle, 2026d).
- `coverage`, which analyzes sentence-level genre coverage across treebanks and can report fully covered versus partially covered inventories, including exportable summaries for regression testing of pattern changes (Stemle, 2026c).

Methodologically, this diagnostic layer is treated as part of the main pipeline rather than as an afterthought. Extraction quality is constrained by (i) metadata availability and (ii) mapping adequacy, and both dimensions vary systematically across languages, genres, and release snapshots. A pipeline that only reports final labels, without systematic coverage accounting, makes it difficult to distinguish between "no genre signal exists" and "a signal exists but is not captured by current patterns."

3.1.4. Extrapolation via weak supervision and clustering

Motivation and relation to prior work. Even with extensive metadata patterning, a substantial fraction of UD data remains multi-genre at the treebank level, without explicit instance-level labels. Müller-Eberstein et al. (2021b) characterise this gap and propose weakly supervised methods for predicting instance-level genre, emphasising that treebank-level metadata alone is a noisy signal that needs disentanglement within treebanks. Related work frames genre metadata as a useful supervisory signal for cross-lingual dependency parsing, in part by projecting treebank-level genre information to sentence level (Müller-Eberstein et al., 2021a).

In this context, we implement a bootstrapped clustering approach aligned with the GMM-based family of methods discussed in this line of work. The core idea is to treat sentence embeddings as a multilingual representation space in which genre-related variation is partially recoverable, and to use existing single-genre resources as anchors for labelling.

Embeddings and clustering. For each UD release snapshot, we generate sentence embeddings from the token `FORM` sequence (i.e. syntactic words), as provided in the Parquet representation. The choice of using token forms rather than reconstructed text is motivated by cross-treebank consistency and by the practical availability of tokenized input in UD. In the current promoted profile, this stage uses multilingual-e5-large with mean pooling over tokenised input and runs identically over local CoNLL-U snapshots and Hugging Face-backed Parquet snapshots.

We then apply clustering at the treebank level. In the current default profile, clustering is performed using Gaussian Mixture Models (GMM), reflecting the need for soft assignments and explicit uncertainty quantification. The number of mixture components is derived from the release-specific genre inventory associated with the treebank, preserving the release-local assumption about how many genres are expected to be present.

Bootstrapped labelling and uncertainty retention. Cluster-to-genre labelling is performed via a bootstrapping mechanism that uses reference material from treebanks that are reliably single-genre (or have sufficiently stable metadata-derived labels). Genre representations are derived from these anchors, and cluster centroids are mapped to genres via similarity-based assignment. Importantly, the pipeline retains uncertainty explicitly rather than forcing hard labels for all instances. In the exported sidecar dataset, this is reflected by separating higher-confidence assignments from lower-confidence inferences, and by shipping row-level provenance (release, treebank, split, configuration, method) together with confidence values, enabling downstream users to trade off coverage against precision in a controlled way.

This design choice follows from the broader observation, already present in prior work, that genre boundaries are not always cleanly separable, and that metadata-derived supervision may be incomplete or ambiguous (Müller-Eberstein et al., 2021b).

3.1.5. Evaluation-oriented validation and release-aware reporting

While the main paper reports results in a dedicated evaluation section, the methodology includes evaluation-oriented validation steps because they influence how we interpret coverage and failure cases.

First, we validate metadata extraction and mapping consistency by comparing predicted labels against available instance-level anchors, where such anchors exist. In the short term, this includes UD-internal signals and the subset of treebanks for which instance-level labels can be reconstructed reliably; in the medium term, we plan alignment against UD-MULTIGENRE, which provides an explicitly enriched UD-based dataset with instance-

level genre annotations and a discussion of inconsistencies in UD genre documentation and usage.

Second, the pipeline supports evaluation regimes that aim to balance comparability with robustness. In particular, our default configuration promotes a strict anchor mode with a training-virtual anchor pool policy, reflecting an attempt to minimise leakage between anchor selection and evaluation targets when assessing generalisation across heterogeneous treebanks.

Third, because our outputs are computed for multiple UD releases, we report coverage and label distributions per release snapshot. This is not only a reproducibility measure, but it also enables identifying when apparent performance changes are better explained by release-driven metadata shifts than by methodological differences.

3.1.6. Publishing as a linked sidecar dataset

A central practical decision is to publish sentence-level genre annotations as a separate dataset on the Hugging Face Hub, linked back to UD sentences through a composite key that includes UD release, treebank identifier, split, and `sent_id`. The key point is that this linkage strategy does not require modifying UD treebanks, and it remains compatible with both Parquet-based workflows and CoNLL-U–based reconstruction. Operationally, each release also carries a frozen configuration snapshot, copied mapping files, and release metadata so that consumers can inspect not only the labels but the exact conditions under which they were produced.

In this framing, sentence-level genre annotation becomes an instance of a more general pattern: attaching derived, task-facing metadata to stable, community-maintained linguistic resources via persistent identifiers. Once this pattern is established, it can be reused to attach additional layers (e.g. domain labels, quality flags, register estimates) in a way that remains compatible with UD's existing governance and release model.

3.1.7. Reference configuration profile

For transparency and reproducibility, we record a default "high-quality" profile that we currently treat as the baseline configuration for generating sentence-level genre annotations.

While the framework is parametrisable, explicitly documenting a default profile serves two functions: (i) it supports replication of the released sidecar dataset, and (ii) it provides a stable reference point for subsequent methodological comparisons, including the planned UD-MULTIGENRE-aligned variant.

4. Evaluation

4.1. Evaluation Orientation

The present work sits between two evaluation targets that should not be conflated. The original GMM+L study on UD v2.8 pools all original test splits into a single 204k-sentence global test partition and reports purity (PUR), agreement (AGR), overlap error (ΔBC), and micro-F1 over the subset of treebanks with instance-level labels (Müller-Eberstein et al., 2021b, Section 5.1 and Table 1). For GMM+L, the published values are PUR $= 1.00$, AGR $= 1.00$, ΔBC $= 0.04$, and micro-F1 $= 0.54$. Our resource, however, is intended for end users who need sentence-level labels that remain useful across new languages, revised treebanks, and changing metadata coverage. We therefore distinguish three notions of parity: *protocol parity* (reconstructing the fixed-partition setup and anchor semantics), *implementation parity* (matching the predictions of an original-like GMM+L pipeline on the reconstructed split), and *end-user evaluation* (sentence-level performance under broader language-held-out conditions).

4.2. Evaluation Regimes

Fixed-partition comparability. To track parity with the original protocol, we reconstruct the paper-style UD v2.8 test partition from the released index split, use whole-treebank single-genre anchors from the same partition, cluster all multi-genre treebanks in that reconstructed test partition, and score only the mapped paper treebanks. This reproduces the paper's evaluation geometry more closely than our broader fixed-partition ablations. Importantly, we still compute sentence-level micro-/macro-F1 on the recoverable sentence-labelled subset, because this is the metric relevant to downstream users; PUR, AGR, and ΔBC are retained as treebank-level comparability diagnostics.

Language-grouped 10-fold generalisation. To approximate the operational setting in which the genre layer is applied to new languages, we additionally run 10-fold cross-validation grouped by *language*. This grouping makes the held-out condition explicit and reduces trivial leakage from shared-language anchors. The evaluation set in this regime is broader ("all_focused"), and the promoted baseline uses multilingual-e5-large embeddings, GMM clustering, and the `combined` anchor policy. The objective is diagnostic and deployment-oriented: to make visible cross-language variance, anchor sparsity, and the cost of stronger coverage assumptions.

4.3. Anchor Construction and Policy Sensitivity

A structural constraint of the UD genre setting is that sentence-level reference labels are not uniformly available. Instance-level "anchor" labels are reconstructed through metadata extraction patterns (comment metadata and `sent_id` conventions), seeded from prior mapping artefacts and iteratively refined using explicit pattern testing and coverage diagnostics. To avoid treating extraction artefacts as model behaviour, we restrict evaluation to treebanks and genres that satisfy strict coverage and minimum-size constraints, and we explicitly report missing-anchor genres.

In the 10-fold regime, we distinguish two anchor policies. The `train_virtual` policy restricts anchors to training-side virtual single-genre subsets, which functions as a conservative baseline and a regression guard. The `combined` policy expands the anchor pool, which can increase coverage and reduce minority-genre sparsity, but also changes the supervision surface and therefore has to be interpreted as a deliberate trade-off rather than as a free improvement.

4.4. Metrics

We report sentence-level and treebank-level metrics separately because they answer different questions. Sentence-level micro-F1 and macro-F1 ask whether individual sentences receive the correct label wherever sentence-level gold can be recovered. This is the primary end-user target: a system can reproduce the overall genre profile of a treebank while still assigning many individual sentences incorrectly.

PUR, AGR, and ΔBC, by contrast, are treebank-level comparability metrics. They ask whether predicted label distributions are internally coherent and whether they align with the expected genre composition of treebanks as specified by metadata. These measures are useful for tracking continuity with Müller-Eberstein et al. (2021b), but they are weaker proxies for the practical utility of a sentence-level annotation layer and can look deceptively strong under collapsed or prior-driven solutions. For that reason, we treat sentence-level F1 as primary and clustering-oriented metrics as secondary diagnostics.

4.5. Quantitative Results

Before turning to the broader end-user results, the parity story can now be stated precisely. On the corrected reconstructed UD v2.8 split, our strict protocol-parity run reaches micro-F1 0.281, macro-F1 0.122, PUR 0.432, AGR 0.625, and ΔBC 0.241 under the current sentence-level evaluator. A direct audit against an original-like reimplementation of GMM+L on the same split yields prediction agreement of > 0.98 on all eight scored treebanks. In other words, once the split reconstruction and anchor semantics are aligned, the remaining gap to the paper's published numbers is not an implementation gap.

Table 1 summarises the three numbers that matter most for interpretation. The first row gives the published GMM+L values from Müller-Eberstein et al. (2021b) under the original paper protocol. The second row gives our reconstructed protocol-parity result under the current sentence-level evaluator. The third row gives the fresh end-user baseline: 10-fold, language-grouped evaluation on the broader UD v2.17 `all_focused` set with multilingual-e5-large, GMM clustering, and the `combined` anchor policy. This baseline yields overall micro-F1 0.333 and macro-F1 0.264; the corresponding fold means are 0.390 ± 0.146 and 0.325 ± 0.163. Unlike the paper-reported row, this baseline is a sentence-level generalisation result on the current broader release rather than a fixed-protocol comparability score.

4.6. Qualitative Confusion Analysis

Confusion-matrix diagnostics, provided as row-normalised heatmaps, suggest that several errors correspond to genre boundaries that are difficult to separate from sentence-bounded context alone. In particular, the broader evaluation runs show systematic cross-talk among *news*, *fiction*, *spoken*, and *nonfiction*, and the conservative anchor policy exhibits a stronger concentration towards dominant assignment pathways. In the fixed-partition setting, uniform reference weighting reduces several collapse patterns observed under sentence-count weighting, which is consistent with the observed improvements in instance-level metrics.

4.7. Relation to Earlier Reported Numbers

A direct numerical comparison with the original paper is therefore informative only if the type of parity is stated explicitly. At the protocol level we recover the paper-style split construction and anchor geometry; at the implementation level we now reproduce the predictions of an original-like GMM+L pipeline exactly on the scored subset. The remaining metric gap arises downstream of the algorithm, chiefly from differences between the paper's treebank-level genre inventories and the sentence-level gold that can be recovered today from the reconstructed subset, as well as from missing-anchor genres in that subset. For example, several paper-expected genres are absent from the recoverable sentence-level gold in the scored portions of Belarusian-HSE,

Regime	Setting	TB	Micro	Macro	PUR	AGR	ΔBC
Paper	GMM+L, UD v2.8 fixed test	–	0.540	–	1.000	1.000	0.040
Protocol parity	Reconstructed UD v2.8 fixed test	8	0.281	0.122	0.432	0.625	0.241
End-user baseline	10-fold by language, UD v2.17 `all_focused`	33	0.333	0.264	0.557	0.592	0.059

Table 1: Published GMM+L values, our reconstructed protocol-parity result, and the current end-user baseline. The protocol-parity run matches an original-like GMM+L implementation on all eight scored treebanks; the end-user baseline has fold means of micro 0.390 ± 0.146 and macro 0.325 ± 0.163. The rows are not directly comparable because they target different evaluation objects.

Czech-CAC, and Russian-Taiga. This makes the fixed-partition parity score a useful comparability instrument, but not by itself the right target for end users.

For this reason, we treat the language-grouped 10-fold evaluation as the main baseline for future improvements. Improvements over the original paper can then be contextualised in two ways: (i) whether they preserve or improve protocol parity, and (ii) whether they improve the more operational sentence-level baseline under held-out language conditions.

4.8. Limitations and Scope

The reported results should be interpreted in the context of three interacting constraints. First, sentence-level genre supervision in UD is sparse and structurally heterogeneous, as only a minority of UD material provides instance-level genre labels in a reusable form, while the majority is annotated at the treebank level and often with multi-genre labels (Müller-Eberstein et al., 2021b). Second, the construction of evaluation anchors relies on metadata extraction patterns and on the stability of sentence identifiers across released resources, which implies that parts of the evaluation set are defined through recoverable documentation conventions rather than through an independently curated gold standard. Third, the language-grouped evaluation highlights that variation across languages and compilation histories is not a marginal effect but a central property of the setting, which limits how far aggregate scores can serve as a single proxy for downstream suitability.

These constraints do not invalidate sentence-level genre as a resource, but they motivate a layered release strategy: predicted labels are most useful when accompanied by provenance, coverage diagnostics, and joinable identifiers that allow consumers to calibrate their own filtering and aggregation choices. The next section, therefore, focuses on the release design and the infrastructural mechanisms by which we make the genre layer reproducible across UD revisions.

5. Outlook: Genre-Aware Evaluation, Adaptation and Reduced Genres

A practical motivation for sentence-level genre annotation is the observation that "high accuracy" preprocessing results are often conditional on evaluation regimes that privilege edited text. In a frequently cited German case study, POS tagging accuracy on Web material drops below the levels reported on standard newswire benchmarks, and the remaining performance varies strongly across Web subgenres (Giesbrecht and Evert, 2009). More recent multi-genre studies in English suggest analogous patterns in social and forum-like text, and also indicate that small amounts of in-domain data can outperform substantially larger amounts of out-of-domain Web data, while ensembles over genre-specialised taggers provide an additional robustness gain (Behzad and Zeldes, 2020; Owoputi et al., 2013b). Taken together, these results suggest that sentence-level genre, when made available as a joinable layer, can support a more transparent separation between (i) model quality under edited conditions and (ii) model robustness under heterogeneous Web and speech-related conditions, without conflating this distinction with language effects.

In this context, two complementary application modes appear plausible. First, genre can be treated as an evaluation variable, enabling stratified reporting for POS tagging, morphological tagging, and parsing, and thereby turning cross-genre variance into an observable quantity rather than an implicit error term. Second, genre can be treated as a selection variable for training, where genre-stratified sampling and targeted proxy selection have already shown benefits in dependency parsing (Müller-Eberstein et al., 2021a; Rehbein and Bildhauer, 2017) and where similar mechanisms have been repeatedly revisited for tagger adaptation in noisy or user-generated settings (Behzad and Zeldes, 2020).

A remaining tension is that the UD genre inventory is relatively fine-grained, partly overlapping, and not always aligned with the text-type distinctions that are commonly used in robustness dis-

cussions. One pragmatic way forward, which we treat as optional rather than as a replacement, is to provide a *reduced* genre layer on top of the full UD inventory. Such a reduction can be designed to align with recurring difficulty profiles, for instance, by separating transcribed speech (*spoken*), user-generated and interactional material (*social*, *email*, and related Web genres), broadly edited Web prose (often approximated by *wiki*), and newswire-like text (*news*). This direction is compatible with existing community efforts to improve cross-treebank genre coherence, such as UD-MULTIGENRE, which proposes a revised set of genres and verified instance-level labels over a UD-derived subset (Danilova and Stymne, 2023).

Finally, sentence-level genre information also connects to more general infrastructure questions around model and data preparation. Domain- and genre-sensitive pretraining has been shown to yield improvements over generic pretraining when the target domain is known (Gururangan et al., 2020). Similarly, tokenizer behaviour is sensitive to the training data and design choices, and recent work suggests that adapting tokenizer training data can affect downstream efficiency and performance (Dagan et al., 2024). In this sense, sentence-level genre labels can be understood as a low-cost coordination layer that enables more explicit experimentation with data composition, rather than as an attempt to impose a single notion of genre on UD.

6. Bibliographical References

Shabnam Behzad and Amir Zeldes. 2020. A Cross-Genre Ensemble Approach to Robust Reddit Part of Speech Tagging. In *Proceedings of the 12th Web as Corpus Workshop*, pages 50–56, Marseille, France. European Language Resources Association.

Gautier Dagan, Gabriel Synnaeve, and Baptiste Roziere. 2024. Getting the most out of your tokenizer for pre-training and domain adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, pages 9784–9805. PMLR.

Vera Danilova and Sara Stymne. 2023. UD-MULTIGENRE – a UD-Based Dataset Enriched with Instance-Level Genre Annotations. In *Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL)*, pages 253–267, Singapore. Association for Computational Linguistics.

Eugenie Giesbrecht and Stefan Evert. 2009. Is Part-of-Speech Tagging a Solved Task? An Evaluation of POS Taggers for the German Web as Corpus. In *Proceedings of the Fifth Web as Corpus Workshop (WAC5)*, pages 27–35, Donostia-San Sebastian, ES.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Max Müller-Eberstein, Rob van der Goot, and Barbara Plank. 2021a. Genre as Weak Supervision for Cross-lingual Dependency Parsing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4786–4802, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Max Müller-Eberstein, Rob van der Goot, and Barbara Plank. 2021b. How Universal is Genre in Universal Dependencies? In *Proceedings of the 20th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2021)*, pages 69–85, Sofia, Bulgaria. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association (ELRA).

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013a. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, Georgia. Association for Computational Linguistics.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013b. Improved Part-of-Speech Tagging for Online Conversational Text with Word Clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–390, Atlanta, Georgia. Association for Computational Linguistics.

Ines Rehbein and Felix Bildhauer. 2017. Data point selection for genre-aware parsing. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 95–105, Prague, Czech Republic.

7. Source Code Documentation References

Max Müller-Eberstein. 2021. ud-genre: Code and artefacts accompanying “How Universal is Genre in Universal Dependencies?”. #5ab79a3, Accessed 2026-03-28.

Egon W. Stemle. 2026a. commul/ud_genre: Sentence-level genre layer for universal dependencies as a separate, joinable dataset. Accessed 2026-03-28.

Egon W. Stemle. 2026b. commul/universal_dependencies: Versioned universal dependencies snapshots in parquet format with release branches and regeneration tooling. v2.2.0, Accessed 2026-03-28.

Egon W. Stemle. 2026c. ud-genre-bootstrap: Command-line interface (coverage diagnostics, evaluation modes). Accessed 2026-03-28.

Egon W. Stemle. 2026d. ud-genre-bootstrap: Genre extraction patterns (documentation). Accessed 2026-03-28.

Egon W. Stemle. 2026e. ud-hf-parquet-tools: Development toolkit for generating and validating universal dependencies datasets in parquet format. v1.2.3, Accessed 2026-03-28.

Universal Dependencies contributors. Universal dependencies: Genres [online]. 2026. Accessed 2026-03-28.